# Amiga Action Replay II

## COMMODORE AMIGA

# DANSK MANUAL.

# INDHOLDSFORTEGNELSE
# AMIGA ACTION REPLAY MKII


**COPYRIGHT .....4**

**MONTERING OG OPSTART ..4**

**VIGTIGT .....4**

**BETJENING .....4**

**SLOW-MOTION .....4**

**FREEZEREN .....5**

**KOMMANDOERNE.....6**

E .....6

**DISKETTEMANIPULATION ...6**

FORMAT .....7
FORMATV .....7
FORMATQ .....7
DISKWIPE .....7
DISKCHECK .....7
DCOPY .....7
CD .....8
DIR .....8
DIRA.....8
MAKEDIR .....8
INSTALL .....8
BOOTPROT .....8
BOOTCODE .....9
DELETE .....9
TYPE .....9
CODE .....9
CODECOPY .....10
RT .....10
WT .....10
DMON .....10
CLRDMON .....10
BOOTCHK .....11
DATACHK .....11
BAMCHK .....11

**FRYSE/MUSIK KOMMANDOER ..... 11**

SA .....11
SR .....12
LA .....12
LR .....12
SLOADER.....12
TRACKER.....12
SCAN .....13

**MEMPEEKEREN ..... 13**

P .....13
SP .....14
SPM .....14

**SPILLETRÆNEREN ..... 15**

**TF KOMMANDOEN..... 17**

**EKSEMPEL PÅ EVIGT LIV .. 17**

**EKSTRA KOMMANDOER ... 19**

RAMTEST .....19
PACK .....19
UNPACK .....19
COLOR .....19
RCOLOR .....19
TMS .....19
TMD .....20
TM.....20
SPR .....20
VIRUS .....20
KILLVIRUS .....20

**MASKINKODEMONITOREN 20**

SETEXCEPT .....20
COMP .....21
SM .....21
SMDC .....21

SMDATA ..... 21
A ..... 21
BS ..... 21
B ..... 21
BD ..... 21
BDA ..... 21
X ..... 22
C ..... 22
D ..... 22
E ..... 22
F ..... 22
FA ..... 22
FR ..... 22
G ..... 23
TRANS ..... 23
WS ..... 23
M ..... 23
N ..... 23
NO ..... 23
NQ ..... 23
MEMCODE ..... 23
O ..... 24
R ..... 24
W ..... 24
Y ..... 24
? ..... 24

**BOOT VIRUSKILLER .....25**

**SYSTEM INFORMATIONER 25**

AVAIL ..... 25
INFO ..... 25
LIBRARIES ..... 25
INTERRUPTS ..... 25
EXCEPTIONS ..... 25
EXECBASE ..... 25
RESOURCES ..... 25
CHIPREGS ..... 26
DEVICES ..... 26
TASKS ..... 26
PORTS ..... 26

**HURTIG GENNEMGANG .....26**

TIPS & TRICKS ..... 27

# AMIGA ACTION REPLAY MKII

# MANUAL PÅ DANSK

## *COPYRIGHT*

Denne manual er omfattet af loven om ophavsret. Den må derfor ikke kopieres, duplikeres eller overføres på nogen måde, hverken helt eller delvist, medmindre der foreligger forudgående skriftlig tilladelse fra Multi Gros A/S og BMP-Data.

# MONTERING OG OPSTART

*For at lette læseligheden af denne manual, er AMIGA ACTION REPLAY MKII i det følgende forkortet til AR.*

På venstre side af din AMIGA 500 finder du en expansionsport. Den kan ikke umiddelbart ses, men er gemt inde bag et lille plasticdæksel, der let kan fjernes. Du bør opbevare plasticdækslet et sikkert sted; det skal ikke bruges, når din AR er i computeren.

På AMIGA 1000 er expansionsporten på højre side af computeren. Husk også i forbindelse med AMIGA 1000, at du skal have KICKSTART ind for at starte op som sædvanlig.

### *VIGTIGT*

VÆR ALTID HELT SIKKER PÅ, AT DIN AMIGA ER SLUKKET, FØR DU MONTERER ELLER DEMONTERER AMIGA ACTION REPLAY MKII! ELLERS RISIKERER DU, AT COMPUTEREN BLIVER BESKADIGET!

## *BETJENING*

AR har tre betjeningsgreb, der alle er meget nemme at anvende. Disse er i det følgende beskrevet.

## *SLOW-MOTION*

To af de tre betjeningsgreb på AR bruges i forbindelse med Slow-Motion funktionen. For at tænde Slow-Motion, skal den lille omskifter sættes i ON-position. At Slow-Motion er aktiveret indikeres med lys i den lille, røde lampe. Når lampen lyser, kan computerens arbejdshastighed reguleres med drejeknappen i hjørnet af AR. Når den er længst mod "+" er hastigheden lavest og vice versa.

Resultatet af brugen af Slow-Motion funktionen kan blive både interessant og meget varieret. Det afhænger fuldstændigt af det program, den anvendes på. I de fleste tilfælde vil den fungere pænt og uforstyrrende, i andre tilfælde kan bevægelserne på skærmen

forekomme ganske ukorrekte og direkte forvirrende. Sidstnævnte gælder specielt ved meget lave hastigheder.

Det er *ikke* anbefalelsesværdigt at bruge Slow-Motion funktionen samtidig med diskettestationen, idet nogle programmer bruger deres egne hurtig-load rutiner, der ikke vil fungere korrekt, når Slow-Motion funktionen er tændt. Men dette giver jo næsten sig selv, da der ikke umiddelbart er nogen grund til at gøre disketteoperationerne langsommere end de er i forvejen.

# FREEZEREN

Den vigstigste funktion i AR er FREEZE-funktionen. På et hvilket som helst tidspunkt i et hvilket som helst program, kan du komme ind i AR's monitorsystem ved et tryk på FREEZE-knappen på AR. Der er dog enkelte programmer, der, grundet deres konstruktion, ikke lader sig fastfryse. Når du trykker på FREEZE-knappen stoppes programmet, der er i gang, totalt. Derefter kommer AR's monitor-skærm frem.

*På dette punkt er det værd at bemærke, at der kan opstå problemer med de danske tastaturer. Først og fremmest er der, i monitor-systemet, ingen danske tegn til rådighed. For det andet kan du komme ud for, at "Z" og "Y" er byttet om. Hvis dette er tilfældet, kan du blot trykke F9. AR husker tastaturindstillingen, så længe computeren er tændt.*

Monitorskærmen er blå med en COPYRIGHT-meddelelse foroven. Denne monitorskærm nås også ved hjælp af breakpoints (forklares senere) samt hvis en virus opdages i computerens hukommelse eller på en diskette.

Når monitorskærmen er fremme, er en del taste-funktioner til rådighed, som er meget nyttige:

| | |
|---|---|
| HELP | Giver en kort beskrivelse af alle kommandoer. |
| SHIFT | Stopper en scrollende skærm midlertidigt. |
| TAB | Indsæt mellemrum. |
| ESC | Afbryder de fleste igangværende funktioner. |
| F1 | Slet skærmen og sæt cursoren i øverste, venstre hjørne. |
| F2 | Sæt cursoren i øverste, venstre hjørne uden at slette skærmen. |
| F3 | Opsætningsskærm (preferences). |
| F5 | Udskriv skærmbilledet på printeren. |
| F6 | Slå printerudskriftsmuligheden til eller fra. |
| F7 | Skift mellem overskriv-/indsæt-måde. |
| F8 | Vis instruktionerne for mempeekeren. |
| F9 | Skift tastatur-nationalitet. |
| F10 | Skift til den alternative skærm. |
| SHIFT F10 | Skift igen til primær skærm. |

Når den røde lampe på computeren (POWER) er slukket betyder det, at AR venter på en kommando eller instruktion.

F3-tasten behøver lidt yderligere forklaring. Når du trykker på F3-knappen, vil en preference-skærm blive fremkaldt. Du vender tilbage til monitorskærmen igen ved tryk på ESC-tasten.

Mulighederne på preference-skærmen vælges/ændres ved hjælp af musen og dennes venstre knap. øverst til venstre illustreres den hukommelse, der er til rådighed for compu-

teren; dette kan ændres ved at klikke på de pågældende kasser og skifte mellem slået til og slået fra (enabled/disabled).

Derunder er valget af boot. Dette anvendes til at vælge, hvilket drev (DF0:, DF1 o.s.v.) computeren skal boote fra. Dette kan ændres helt frit eller sættes til "variabelt", så at maskinen booter fra et hvilket som helst drev, der indeholder en diskette, som man kan boote fra.

Nederst til venstre vælges AR farverne. De to skalaer i højre side af preference-skærmen anvendes til at vælge hastighed for den automatiske skydefunktion for de to joysticks. De kan indstilles helt uafhængigt af hinanden, hvilket giver mulighed for at give en god spiller et handicap. Den automatiske virus-detektor kan også slås fra via denne skærm.

# KOMMANDOERNE

I det følgende er alle de kommandoer, der kan anvendes i monitorsystemet, gennemgået. Ved mange af kommandoerne kan du angive ét eller flere parametre efter selve kommandoen. Dette giver dig uanede muligheder, der selvfølgelig ikke alle er forklaret til bunds i denne manual.

Kommandoerne er opdelt i logiske funktionsgrupper. De detaljer, der står i trekantede paranteser (<>) er parametre, hvoraf nogle af dem kan udelades. Tal kan angives både decimalt, hexadecimalt og binært. AR's logiske base er det hexadecimale talsystem og altså IKKE det decimale talsystem, som vi er vant til at anvende.

Hvis du ikke er vant til at angive tal hexadecimalt, skal du huske, at alle tal, du angiver som parametre i AR, skal have et udråbstegn (!) foran, idet ! er præfix for decimale talangivelser i AR. Ligeledes er % præfix for binære talangivelser.

Herunder vises en kommando fra manualen som et eksempel

***E <offset>***

Dette betyder en kommando, der hedder E, som skal efterfølges af et parameter, der i dette tilfælde er et tal. Hvis parametret 32 ønskes, skal der altså enten skrives:

E !32

eller

E 20

Alle kommandoer skal efterfølges af et tryk på RETURN-tasten.

# DISKETTEMANIPULATION

De følgende kommandoer kræver normalt, at man har Workbench startet op og i gang, for over hovedet at kunne udføre dem. Men, når du har AR i computeren er kommandoerne altid til rådighed, også selv om et program er i gang i hukommelsen. Du kan bruge dem simpelt hen ved at trykke på FREEZE-knappen og trykke dem ind. Hvis intet andet specificeres, arbejder funktionen på det i øjeblikket aktive drev. Du kan naturligvis angive

drev-nummer og sub-directory som i CLI - se AMIGA'ens manual for nærmere information herom.

### *FORMAT <navn>*

Denne instruktion formaterer en diskette til standard Amiga DOS format (det, der normalt anvendes) med det angivne navn. Denne funktion sletter alle de data, der måtte befinde sig på disketten, så brug den med omtanke! Den skal til gengæld anvendes på alle nye disketter, før disse kan tages i brug. Eksempel:

FORMAT DEMO

hvortil computeren svarer:

READY TO FORMAT DISK IN DRIVE DF0:

hvilket du bekræfter ved at trykke på Y og RETURN. Nu formateres den diskette, der ligger i DF0: (det interne diskettedrev) og den bliver kaldt DEMO.

### *FORMATV <navn>*

Bruges på samme måde som ovenstående FORMAT-kommando. Forskellen er imidlertid, at AR nu tjekker disketten, når den er formateret, for evt. fejl. Det tager lidt længere tid, men til gengæld opdager du fejl inden det er for sent.

### *FORMATQ <navn>*

Denne kommando bruges også til formatering efter samme syntax som de to ovenfor. Den kan kun anvendes på disketter, der har været anvendt før. Forskellen er, at det går stærkt. Meget stærkt!

### *DISKWIPE <drevnr.>*

Sletter alle informationer på disketten. Herefter kan disketten ikke anvendes, før den er blevet reformateret med enten FORMAT eller FORMATV. Eksempel:

DISKWIPE DF0

vil slette alle data på den diskette, der ligger i drev DF0:.

### *DISKCHECK <drevnr.>*

Denne instruktion vil scanne alle spor på en diskette i det angivne drev og vil angive på skærmen, hvis den finder nogle fejl på disketten. Alle spor, der ikke er formateret i AMIGA-DOS formatet, vil også blive angivet som fejl.

### *DCOPY <drevnr. for læsning> <drevnr. for skrivning>*

Denne kommando kopierer en AMIGA-DOS diskette (sædvanligvis de fleste disketter) fra et drev til et andet som angivet med parametrene efter DCOPY. Du kan angive det samme drevnr. to gange (f.eks. DCOPY DF0 DF0), men dette vil slette alle programdata i computerens hukommelse. Disketten, der skal skrives på, behøver ikke at være formateret. Hvis der er noget på den i forvejen, vil det blive slettet under kopieringen. Somme tider kan kopieringen forårsage fejl på kopien, hvis der som kopidiskette anvendes en diskette, der har været brugt tidligere. Eksempel:

DCOPY DF0 DF1

Her kopieres den diskette, der er i DF0, over på disketten i DF1.

### *CD <path>*

Kommandoen uden path-angivelse vil vise det nuværende drevnr. samt evt. subdirectory (path). Hvis du vil ændre path, gøres det med standard CLI-instruktioner (se evt. AMIGA-manualen). Eksempel:

CD QWERTY/SUBDIR

vil ændre det nuværende arbejdsområde (path) til sub-directoriet SUBDIR, der findes i directoriet QWERTY.

CD /

vil gå tilbage til det forrige directory.

### *DIR <path>*

Denne kommando giver en liste over de filer og sub-directories, der er i det nuværende eller angivne path på disketten. Et path kan også angives for at se indholdet i et bestemt område på disketten. Eksempel:

DIR QWERTY/TWO

vil uden hensyntagen til aktivt path liste indholdet af subdirectoriet TWO, der findes under QWERTY.

### *DIRA <path>*

Svarer til ovenstående DIR-kommando med den undtagelse, at indholdet af alle sub-directories også bliver listet, så du med denne kommando får en komplet oversigt over indholdet. Hvis skærmen ruller inden du når at aflæse den, kan den stoppes midlertidigt med SHIFT.

### *MAKEDIR <path>*

Med denne instruktion kreéres et nyt subdirectory i det nuværende path/aktive directory, såfremt det ikke allerede findes. Eksempel:

MAKEDIR SUB1

laver et subdirectory, der hedder SUB1, i det aktive directory.

MAKEDIR MAIN/SECOND/SUB1

laver subdirectoriet SUB1 i subdirectoriet SECOND, der findes i hoveddirectoriet MAIN.

### *INSTALL <bootblock nr.>*

Med denne kommando kan du installere en boot-block i boot-sektoren i det aktive drev. Dette giver dig samtidig muligheden for at afvikle frosne programmer (se senere!) uafhængigt af AR. Her kan du også få brug for SLOADER-kommandoen. Installationen foregår simpelt hen ved at indsætte disketten, skrive INSTALL og trykke RETURN.

### *BOOTPROT <kodenr.>*

Denne kommando åbner mulighed for at beskytte bootblokken på en diskette med et ottecifret tal (eller mindre), hvilket gør det umuligt for folk, der ikke kender koden, at boote disketten. Til gengæld skal BOOTCODE kommandoen (forklaret herunder) anvendes til at bruge disketten igen - en ekstra beskyttelse af dine vigtigste disketter! Det er ikke anbefalelsesværdigt at kode en diskette mere end én gang. Eksempel:

BOOTPROT 1234

For at være helt sikker, spørger AR:

SURE TO PROTECT DRIVE IN DF0:

Hvortil du kan acceptere ved at trykke Y og RETURN. Når dette er gjort, beskyttes disketten med koden 1234, som du skal skrive op eller huske (hvis du glemmer den, er disketten tabt, så det er virkelig en god idé at huske koden!).

### *BOOTCODE <kodenr.>*

BOOTCODE-kommandoen er den eneste vej til at boote en diskette, der er blevet kodet med BOOTPROT-kommandoen (se ovenfor). Hvis du vil boote en beskyttet disk, skal du først trykke på FREEZE-knappen, derefter trykke BOOTCODE efterfulgt af diskettens kode og et tryk på RETURN. Eksempel:

BOOTCODE 1234

Med denne kommando kan den diskette, der blev kodet med BOOTPROT 1234 (se ovenfor), bootes op igen. Resten af bootningen forløber som normalt, dog skal du lige trykke X for at vende tilbage til programmet. Vær opmærksom på, at AR husker den senest angivne bootkode, i dette tilfælde 1234. Derfor kan du komme ud for, at en ikke kodet diskette ikke vil boote, hvis du tidligere har anvendt en bootcode uden at slette den igen. Derfor kan BOOTCODE 0 bruges til at komme af med koden. Hvis du vil se den senest anvendte boot-kode, er det nok at skrive BOOTCODE uden parameter.

### *DELETE <path> <navn>*

Med DELETE-kommandoen slettes en fil på en diskette. Hvis intet path angives, søges filen i det aktive path/directory. Eksempel:

DELETE WAYNE

Denne kommando fjerner filen WAYNE fra det aktive directory, såfremt det findes.

DELETE MAIN/SUB/WAYNE

Fjerner filen WAYNE i det aktive subdirectory SUB i hoveddirectoriet MAIN.

### *TYPE <path> <navn>*

Her vises indholdet af den angivne fil på skærmen som ASCII karakterer. Et program, der kan afvikles, kan forekomme som rent affald på skærmen, hvorimod data fra en tekstbehandling i det store hele viser den indtastede tekst samt nogle ekstra karakterer, der styrer f.eks. sideskift, understregning, kursiv o.s.v. Path og filnavn angives som ved DELETE.

### *CODE <drevnr.> <kodenr.>*

Dette er endnu en ret kompleks krypteringskommando, der kræver lidt erfaring for at opnå det rette udbytte. <drevnr.> er drev nummer 0-4 og <kodenr.> er en værdi fra 0-!65535. Den letteste måde at demonstrere denne kommandos effekt på, er ved eksempler:

CODE 0 3

Al datastrøm til drev DF0: vil herefter blive kodet med værdien 3, så at de kun kan læses korrekt igen ved at blive dekodet med værdien 3. CODE-kommandoen sætter samtidigt dekode-værdien. Det betyder, at al datastrøm fra drev DF0: samtidigt bliver dekodet med værdien 3. Det vil få normale filer til at blive indlæst som nonsens. Eksempel:

CODE 0
fjerner kodning og dekodning fra drev DF0:.

SA TEST
gemmer en frossen fil på diskette (SA forklares senere).

CODE 0 3
koder og dekoder datastrøm til/fra drev DF0: med 3.

LA TEST
henter den frosne fil fra disketten igen. Men nu vil den ikke virke, da den bliver dekodet med 3, hvorimod den blev gemt uden kodning.

CODE
uden parametre viser alle de kodede/dekodede drev og det brugte kodenr. Bemærk, at dette kun er AR's hukommelsestilstand og altså ikke kan bruges til at aflæse kodenr. på disketterne. De numre, der angives, svarer IKKE til de indtastede, f.eks. vil du af det følgende se, at code 000000005 svarer til det indtastede CODE 0 3. Således vil 000000005 altid svare til 3! Eksempel:

```
ACTION REPLAY DISK CODER V1.1
=======================================
DRIVE0 CODE:000000005 PROTECTED
DRIVE1 CODE:000000000 NORMAL
DRIVE2 CODE:000000000 NORMAL
DRIVE3 CODE:000000000 NORMAL
DRIVE4 CODE:000000000 NORMAL
```

HUSK at bruge denne kommando med stor omtanke! Husk kodenumrene, når du har kodet et program, og lad være med at bruge den på alt for vigtige data. Koden kunne jo blive væk!

### *CODECOPY <drevnr. for læsning> <drevnr. for skrivning>*

Med fantasirig brug, kan denne kommando bruges til mange forskellige ting. Den kopierer en hel diskette ved først at dekode med det kodenr., der hører til <drevnr. for læsning> og derefter kode med det kodenr., der hører til <drevnr. for skrivning>. Dette kan f.eks. bruges til at dekode en hel diskette eller til at lave en sikkerhedskopi, som andre ikke kan bruge, hvis de får fingre i den.

### *RT <spornr.> <antal> <adr.>*

Læser spor fra disketten startende i <spornr.> og <antal> halve spor frem, og skriver disse i hukommelsen startende i den adresse, der er angivet med <adr.>. Hvis <adr.> ikke er angivet, vil computeren forsøge at finde et område i hukommelsen til diskettedataene. Læg mærke til, at antallet af spor er halve spor, f.eks. tæller spor 0 side 0 som ét halvt spor, mens spor 0 side 1 tæller som et andet!

### *WT <spornr.> <antal> <adr.>*

Svarer til RT, blot skrives her de tilsvarende spor fra hukommelsen på disketten. Angives <adr.> ikke, gemmes bufferen.

### *DMON*

Angiver hvor i hukommelsen, læste spor er, hvis de er læst med RT.

### *CLRDMON*

Sletter diskettemonitorens bufferområde, der automatisk blev oprettet med RT-kommandoen. Det er en god idé at anvende CLRDMON før RT-kommandoen, når du undersøger

og arbejder med en diskette med RT- og WT-kommandoerne, da CLRDMON søger for, at sporene ikke bliver mixet sammen.

### *BOOTCHK <sektoradr.>*

Denne kommando bliver brugt på en sektor, der er blevet læst ind i hukommelsen med RT kommandoen. Den tjekker checksummen på en bootblock, der starter i hukommelsen ved <sektoradr.>. Hvis der findes en checksum-fejl i bootblocken, vil den blive rettet.

### *DATACHK <sektoradr.>*

Svarer til BOOTCHK-kommandoen (se ovenfor), blot tjekkes her checksummen på en datablok. Evt. fejl vil blive rettet.

### *BAMCHK <sektoradr.>*

Ligeledes tjekkes her checksummen på en BAM-sektor, der er hentet ind i hukommelsen med RT-kommandoen. Evt. fejl i checksummen vil blive rettet.

# FRYSE/MUSIK KOMMANDOER

Når du trykker på FREEZE-knappen, fastfryses alt, hvad der foregår i computeren, og kun monitorsystemet i AR er aktivt. På grund af en lille RAM-udvidelse i AR koster monitorsystemet ikke noget af hukommelsen i computeren, hvorfor alle registre og den totale hukommelse forbliver urørt, når programmet startes igen med X-kommandoen. Dette giver imidlertid nogle ekstra fordele, som bl.a. ses i kommandoerne i dette afsnit.

### *SA <path> <navn>,<kompressionsgrad>*

SA bruges til at gemme den totale hukommelse med samtlige registre på diskette. Dette kaldes for en frossen udgave af programmet eller bare en frysefil. Som nævnt under DELETE (se forrige kapitel) angives <path> og <navn>, hvor <path> kan udelades. <navn> er det navn, som du vil give frysefilen. <kompressionsgrad> angiver, i hvor høj grad, programmet ønskes pakket sammen. Tallet kan være fra 0 til !65535; jo højere tal, des mindre plads vil frysefilen tage på disketten. Imidlertid vil en stor kompressionsgrad betyde lang kompressionstid. Kompression ved maksimal kompressionsgrad vil tage meget lang tid. Som regel vil en værdi omkring !200 være passende. Eksempel:

SA TEST
SA TEST1,!50
SA TEST2,!200
SA TEST3,!400

Dette vil efter DIR give dette resultat (tallene kan variere, men forholdet vil være omtrent det samme):

184210 TEST
145862 TEST1
091226 TEST2
084690 TEST3

Bemærk, at kompressionen bliver mindre og mindre effektiv for hver gang, kompressionsgraden ændres, også selv om kompressionsgraden fordobles!

### *SR <path> <navn>,<kompressionsgrad>*

Identisk med SA med den ene forskel, at programmet genstartes, så snart det er gemt på disketten.

### *LA <path> <navn>*

Denne funktion vil hente en frysefil fra disketten. I praksis den modsatte funktion af SA. Efter loadningen er programmet klar til at blive startet med kommandoen X. Programmet vil fortsætte derfra, hvor det blev frosset fast. Eksempel:

LA TEST2

### *LR <path> <navn>*

Identisk med LA med den ene forskel, at programmet automatisk genstartes, når det er loadet ind.

### *SLOADER*

Hvis du ønsker, at en frysefil skal kunne startes uafhængigt af AR, skal du bruge SLOADER-kommandoen til at lægge et ekstraprogram på disketten. Først skal INSTALL-kommandoen bruges og herefter SLOADER. Når disketten herefter sættes i, vil den autoboote og vente på en kommando. På dette punkt kan kommandoen ALOAD bruges. ALOAD skal efterfølges af programmets navn. Eksempel:

ALOAD TEST2

ALOAD-kommandoen skal selvsagt ikke anvendes på monitorskærmen i AR (efter FREE-ZE), men kan anvendes helt uden AR i computeren.

### *TRACKER*

Ved brug af denne kommando vil AR søge hele hukommelsen igennem for at finde musikstykker, der er konstrueret med et såkaldt tracker-program. TRACKER-funktionen fungerer bedst på Public-Domain software, hvor forfatterne i højere grad bruger færdige musikpakker (tracker-programmer) til at lave musikken med.

Vær opmærksom på, at TRACKER-kommandoen ikke er et mirakel, der kan finde alle slags musikstykker i alle spil/programmer. Musik har ikke noget standardformat ligesom samples og grafikbilleder har det. Mange komercielle programmer anvender deres eget format, som TRACKER'eren af gode grunde ikke kan finde.

Skærmen vil blinke med forskellige kombinationer af farver, mens AR søger hukommelsen igennem for hver type tracker-format. Når AR finder et stykke musik, vil flere muligheder og oplysninger blive vist på skærmen.

Det første tal er startadressen for det fundne musikstykke i computerens hukommelse. Sangtypen (songtype) angiver navnet på det tracker-program, som musikken er lavet med. Sangnavnet (songname) er det navn, musikstykket fik, da det blev skrevet i tracker-programmet.

Funktionstasterne har i dette tilfælde følgende funktioner:

| | |
|---|---|
| F1 | Afspil musikstykket. |
| F2 | Stop dette musikstykke. |
| F3 | Vis flere detaljer/informationer. |
| F4 | Gem musikstykket på diskette i det angivne format. |
| F5 | Giv musikstykket et nyt navn. |
| F6 | Vis sangdataene, brugbar ved afspilning. |

| | |
|---|---|
| F7 | Fortsæt søgning efter nye musikstykker. |
| F8 | Forsøger at ændre samplet til Soundtracker med 16 samples. |
| F9 | Vil opdatere visse af dataene. |
| F10 | Stop TRACKER-funktionen. |

Læg mærke til, at et stykke musik vil blive ved med at spille, hvis det er sat i gang lige før TRACKER-funktionen stoppes. Det lyder godt, men lad være med at forsøge på at formatere eller kopiere en diskette, mens musikken spiller. Det kan få computeren til at crashe!

### *SCAN*

SCAN bruges til at finde samplede lyde i hukommelsen. En samplet lyd kan f.eks. være tale i et spil (f.eks. HUGO: "Hvor ska' vi hen, du?"). Efter valg af SCAN-funktionen, vil AR vise en skærm med en graf over hele computerens chipmem, der bliver vist som ét stort sample. Opgaven er nu at finde det ønskede sample i hukommelsen, det vil sige at indkredse det.

Følgende F-taster er til rådighed:

| | |
|---|---|
| F1 | Afspil sample mellem de to markører. |
| F2 | Opdatér skærmen, vis området mellem de to markører. |
| F3 | Sætter markeringen til fuld størrelse og opdaterer skærmen. |
| F4 | Udvider det markerede område og opdaterer skærmen. |
| F5 | Gemmer området mellem markørerne på diskette som IFF-fil. |
| Space | Skifter aktiv markør mellem start/slut. |
| Venstrepil | Flyt aktiv markør mod venstre. |
| Højrepil | Flyt aktiv markør mod højre. |
| Oppil | Forlæng sampletiden. |
| Nedpil | Forkort sampletiden. |

For at finde et sample i hukommelsen, spilles først hele hukommelsen med F1. Læg mærke til, hvor +-tegnet er, når det sample, du skal finde, bliver afspillet. Flyt start-markøren til venstre side af området og slut-markøren til højre side. Opdatér skærmen med F2 og afspil igen. Fortsæt på denne måde indtil du har fundet de korrekte grænser for samplet. ændr evt. på sampletiden med op-/ned-pilene. Når det hele er i orden, kan samplet gemmes på diskette med F5.

# MEMPEEKEREN

Mempeekeren er en helt unik mulighed for at ændre helt frit på et frossent skærmbillede. Billedet kan komme hvor som helst fra, f.eks. fra et spil. Til mempeekeren er knyttet flere kommandoer, der i det følgende er gennemgået.

### *P <billednr.>*

Når der er trykket RETURN efter indtastningen af denne kommando, vil skærmbilledet fra frysningen vise sig på skærmen. Samtidig kommer en masse kommandoer inden for rækkevidde. Disse er kommandoer til manipulation af det frosne skærmbillede. Oversigten over disse særlige mempeeker-kommandoer kan fås på skærmen ved tryk på F8, men de er herunder kort gennemgået på dansk.

Efter P er det ikke nødvendigt at angive et billednr. I særlige tilfælde kan dette dog angives.

De effekter, der kan opnås ved kombination af disse fleksible kommandoer, vil bruge flere bøger at beskrive til bunds på grund af AMIGA-grafikkens store kompleksitet. For at lære resultaterne at kende, er kun én metode virkelig effektiv: Prøv dig frem! Du behøver ikke at vide, hvad de tekniske ord, der er knyttet til hver enkelt kommando, betyder. Det kan aldrig skade at prøve sig frem og tjekke resultatet - det kan i mange tilfælde være ganske slående!

| | |
|---|---|
| a | automatisk plan-funktion |
| b | forøg belysning |
| SHIFT b | formindsk belysning |
| c | forøg farveregister |
| d | dual playfield slået til |
| SHIFT d | dual playfield slået fra |
| e | forøger højre kant |
| f | et plan op hurtigt |
| SHIFT f | et plan ned hurtigt |
| h | HAM (4096 farver) slået til |
| SHIFT h | HAM (4096 farver) slået fra |
| i | invertér alle farver |
| l | lavopløsningsmåde (LORES) |
| SHIFT l | højopløsningsmåde (HIRES) |
| m | forøg modulo-effekt |
| n | formindsk modulo-effekt |
| s | forøger venstre kant |
| SHIFT s | formindsker venstre kant |
| w | hvid hjælpeskærm |
| SHIFT w | sort hjælpeskærm |
| x | formindsk farve-register |
| 0 | løsn alle planer |
| SHIFT 0 | fastlås alle planer |
| 1 | fastlås plan 1 |
| SHIFT 1 | løsn plan 1 |
| 2 | fastlås plan 2 |
| SHIFT 2 | løsn plan 2 |
| 3 | fastlås plan 3 |
| SHIFT 3 | løsn plan 3 |
| 4 | fastlås plan 4 |
| SHIFT 4 | løsn plan 4 |
| 5 | fastlås plan 5 |
| SHIFT 5 | løsn plan 5 |
| 6 | fastlås plan 6 |
| SHIFT 6 | løsn plan 6 |
| 7 | fastlås plan 7 |
| SHIFT 7 | løsn plan 7 |
| 8 | fastlås plan 8 |
| SHIFT 8 | løsn plan 8 |
| 9 | fastlås plan 9 |
| SHIFT 9 | løsn plan 9 |
| F1 | Fastsæt farver |
| F10 | Tilfældige farver |
| venstrepil | rotér billedet mod venstre |
| højrepil | rotér billedet mod højre |
| pil op | rul billedet opad |
| SHIFT pil op | rul billedet hurtigt opad |
| pil ned | rul billedet nedad |
| SHIFTpilned | rul bileldet hurtigt nedad |
| Delete | gem hjælpeskærmen |
| ESC | stop mempeekerfunktionen |
| HELP | vis hjælpeskærmen |
| v. museknap | forøg billedhøjden |
| h. museknap | formindsk billedhøjden |

Hjælpeskærmen rokeres rundt ved at flytte musen.

### *SP <path> <navn>,<billednr.> <højde>*

SP bruges til at gemme billedet, som det er i det frosne program og altså IKKE med de ændringer, der er foretaget ved hjælp af mempeekeren. <path> og <navn> angives som vist tidligere i denne manual. <billednr.> og <højde> er billednummeret og billedets højde. Hvis du f.eks. har fastfrosset kickstart-skærmen (den med hånden) og gerne vil gemme den på diskette som en IFF-fil, kan dette gøres med:

SP KICKSTART,1 !232

Når dette er gjort, kan billedet hentes ind i f.eks. Deluxe Paint eller Photon Paint og redigeres som du har lyst til. Ofte vil du anvende den store højde og billedet er nr. 1. I så tilfælde kan du nøjes med at skrive som i følgende eksempel:

SP TESTPIC

### *SPM <navn>*

Med SPM-kommandoen gemmes skærmen med de ændringer, du måtte have foretaget med mempeekeren, f.eks. ændringer i farven. Denne feature er idéel til at gemme

skærmbilleder fra spil med det formål at bruge et tegneprogram til at editere og/eller udskrive det med.

I det følgende er et meget kort eksempel på, hvorledes mempeekeren kan bruges. Eksemplet er baseret på kickstart-hånden, der kommer, når du tænder computeren. Start altså med at tænde computeren og tryk så på FREEZE-knappen, når hånden står på skærmen. Når den blå monitorskærm fremkommer, skriver du følgende kommandoer:

P (+ RETURN).
Hermed startes mempeekeren.

HELP
Hjælpeskærmen tændes.

SHIFT + w
Gør hjælpeskærmen synlig.

Venstre museknap
For at reducere skærmstørrelsen til den rette højde.

9
Læg mere grøn farve på.

ESC
Stop mempeekeren

SPM KICKSTART
gemmer det ændrede KICKSTART-billede på diskette, der er grønnere end det "gamle".

# SPILLETRÆNEREN

Spilletræneren er den del af AR, der giver dig mulighed for at få flere liv i de fleste spil. Princippet går ud på, at AR forsøger at finde de adresser i computeren, der indeholder antallet af tilbageværende liv. Dette gøres ved i første omgang at oplyse AR om antallet af tilbageværende liv (TS-kommandoen), dernæst miste et liv og lade AR indkredse de adresser, der har fulgt den samme udvikling (T-kommandoen). Når denne indkredsning således har ramt én adresse, kan man være ret sikker på, at den pågældende adresse tæller antallet af tilbageværende liv. Derfor bliver det på denne måde direkte muligt at ændre dette antal.

Alle kommandoer, der har noget med spilletræneren at gøre, starter med T. Følgende tre kommandoer bruges til at give ekstra liv:

a) TS <værdi>
b) T <værdi>
c) TX

TS starter spilletræneren og leder efter den angivne værdi (antal tilbageværende liv). Med TS/T kan du få et tal, der angiver adressen i hvilken antallet af liv er angivet. Ved hjælp af denne information kan du med M kommandoen (se længere fremme i manualen) enten forøge eller formindske antallet for at skabe flere liv.

T fortsætter spilletræneren med den angivne værdi (forskellig fra den, der tidligere blev angivet med TS). Dette kan f.eks. være, hvis du har mistet et liv i spillet i mellemtiden. På denne måde kan AR bedre indkredse antallet af mulige adresser. Hvis spilletræneren ikke

kan finde en adresse, der har haft denne udvikling, vil den fortælle, at det gik galt. I dette tilfælde kan du enten prøve at starte spilletræneren igen med TS eller bruge TX for at komme ud af spilletræneren.

TX stopper spilletræneren og frigører den hukommelse, spilletræneren optager i AR's indbyggede hukommelses-udvidelse.

Eksempel:

TS!3
FIRST TRAIN PASS
CHANGE COUNTVALUE NEXT TIME
SEARCH UPTO: 005444
TRAIN MODE ACTIVE
READY

Start spillet op igen med X og RETURN. Spil indtil netop endnu ét liv er brugt. Brug igen FREEZE-knappen for at komme ind i monitorsystemet. Brug T-kommandoen til at lede efter de adresser, der blev fundet med TS, med det nye antal liv.

Eksempel:

T!2
POSSIBLE ADDRESSES:
00014428
SEARCHED UPTO :080000
TRAINMODE ACTIVE
READY

Her er altså 14428 (hex) en mulig adresse. Der kan være flere muligheder. I så tilfælde skal spillet startes igen og endnu et liv bruges, så at der i forhold til eksemplet kun er ét liv tilbage. Derefter bruges kommandoen T!1. På denne måde indkredses, til kun én mulig adresse (possible addresses) vises på skærmen - denne er da ganske sikkert den adresse, der indeholder antallet af liv.

I første omgang kan du forsøge at få evige liv med TFD-kommandoen, der er nærmere forklaret lidt længere fremme:

TFD 14428

Såfremt computeren bl.a. svarer med SUB FOUND AT ..... SUBS ELIMINATED, vil du herefter have evigt liv! Spillet genstartes i så tilfælde med X og RETURN. Hvis ikke, må du ændre værdien i adresse 14428 manuelt med M-kommandoen, der er nærmere forklaret i afsnittet om maskinkodemonitoren. Fremgangsmåden er i vores eksempel at skrive:

M14428

Nu viser skærmen indholdet i adresse 14428 og de derpå følgende syv adresser. Placér cursoren over det første tal, hvor der formodentlig står anført antallet af tilbageværende liv. Skriv nu det ønskede antal liv (angives hexadecimalt, d.v.s. max. 2 cifre), f.eks. F0 (svarer til 240 liv). Tryk RETURN og ESC. Herpå er det nok blot at trykke X og RETURN for at komme tilbage til spillet. Dog er det en god idé at notere adressen (i dette tilfælde 14428), da den ofte vil være den samme, næste gang spillet startes. På denne måde slipper du for at gå igennem hele spilletræneren en gang til. Dette gælder også, hvis det lykkes dig at få evigt liv med TFD-kommandoen.

Nogle spil kan ikke arbejde med store mængder af liv og kan derfor crashe, når et stort antal liv vælges med M. Andre kan vise en slags grafik-affald dér, hvor antallet af liv normalt

står. Dette er dog ingen hindring for, at spilletræneren kan virke udmærket alligevel. I de spil, der ikke kan arbejde med de store antal liv, er det muligt at prøve med lidt mere beherskede antal, f.eks. 9 liv (09).

Somme tider kan du komme ud for, at spillet ikke med det samme viser det nye antal liv - det kan være, at det først viser sig næste gang, et liv er mistet, når spillet kommer til en ny bane eller lignende.

På samme måde som med livene, kan enhver kendt værdi for ammunition, liv, brændstof eller lignende enten forøges eller formindskes på denne måde.

Det er ikke alle spil, der kan samarbejde med spilletræneren. For ekstra hjælp i de besværlige: Se kapitlet sidst i denne manual med Tips & Tricks.

## *TF kommandoen*

TF er den absolutte spilletræner. Den kræver et vist kendskab til maskinkodeprogrammering for at kunne blive udnyttet fuldt ud. Den giver mulighed for at få evigt liv, idet den kan påpege hvor i et program, den med de ovennævnte kommandoer fundne adresse bliver formindsket. På denne måde kan denne kommando erstattes af en mere harmløs. Følgende to udgaver findes:

a) TF <adresse>
b) TFD <adresse>

TF adresse leder hukommelsen for at finde den pågældende SUBI-/SUBQ-kommando, da det højst sandsynligt er én af disse, der anvendes til at formindske antallet af liv.

TFD fungerer på samme måde som TF med den forskel, at den sørger for at fjerne disse kommandoer og erstatte dem med nogle mere harmløse - kort og godt til at få evige liv!

## *Eksempel på evigt liv*

Hvis du er forvirret oven på hele denne lidt tekniske snak om spilletræneren, så prøv en gang at følge dette eksempel på, hvordan du kan få flere liv. Eksemplet er udarbejdet ud fra det populære spil "SuperOswald".

Først loades spillet på helt normal vis. Det startes som sædvanligt. Eksemplet er lavet ud fra "EN SPILLER" valget. De angivne adresser kan variere, men fremgangsmåden er den samme.

Når spillet er i gang, trykkes på FREEZE-knappen, hvorpå monitor-systemet træder i kraft. Tryk nu følgende:

TS!5

og tryk RETURN. Femtallet angiver, at der er fem liv tilbage. Herpå svarer computeren følgende:

FIRST TRAINPASS!
CHANGE THE COUNTVALUE NEXT TIME!
SEARCHED UP TO: 0004E8B0
TRAINMODE ACTIVE!
READY

Tryk nu X og RETURN for at komme tilbage til spillet. Sørg for at miste ét liv og vent, til det nye antal liv stør på skærmen (4), således at du er helt sikker på, at computeren har opdateret antallet af tilbageværende liv. Tryk nu atter på FREEZE og skriv dette:

T!4

Efter tryk på RETURN svarer computeren:

POSSIBLE ADDRESSES:
0000A221 00020DB6
SEARCHED UP TO: 00C04336
TRAINMODE ACTIVE!
READY

Dette betyder, at der nu er to muligheder. Da der kun skal bruges én adresse, må sidste trin gøres igen. Det vil sige, at spillet atter startes, ét liv mistes. "Det var vel nok ærgerligt" overstås og tretallet kommer på skærmen. Tryk igen på FREEZE og skriv følgende:

T!3

Hvorefter computeren svarer:

POSSIBLE ADDRESSES:
0000A221
SEARCHED UP TO: 00C1A08E
TRAINMODE ACTIVE!
READY

Hermed er adressen fundet! At der kun er én mulig adresse (possible addresses) betyder, at netop dén adresse, der indeholder antallet af tilbageværende liv, er indkredset. I vores SuperOswald-eksempel er den søgte adresse altså A221 (hexadecimalt angivet).

For at få flere liv skrives nu:

M A221

til hvilket computeren skriver:

:00A221 03 00 01 FF F6 00 00 00 .....

Tryk nu følgende taster for at få 240 liv:

F 0 RETURN ESC X RETURN

Det andet tegn er et nul. På denne måde startes spillet samtidigt. Næste gang antallet af liv opdateres på skærmen, vil antallet af tilbageværende liv være erstattet af en grafikrubrik uden fornuftigt indhold. Dette har intet at sige for funktionen af spillet. Grunden er den, at tallet 240 ikke kan skrives med ét ciffer.

Da adressen kan være god at have til en anden gang, er det en god idé at skrive den op. På denne måde kan du, næste gang, du vil have flere liv, blot trykke FREEZE og skrive følgende:

M A221

og derefter trykke på følgende taster:

RETURN F 0 RETURN ESC X RETURN

Iøvrigt kan det samme gøres med f.eks. antallet af gaver i SuperOswald.

# EKSTRA KOMMANDOER

### *RAMTEST <startadr.> <slutadr.>*

Tjekker RAM-hukommelsen for evt. fejl. Det angivne hukommelsesområde vil blive slettet for at udføre testen. Først skrives nuller i alle adresser og derefter FF (d.v.s. alle bits bliver først sat til 0, derefter til 1). I begge tilfælde tjekkes efter, om værdierne kan aflæses korrekt derefter. Evt. fejl vil blive rapporteret.

### *PACK <startadr.> <slutadr.> <dest.adr.> <kompressionsgrad>*

Denne kommando komprimerer en angiven del af hukommelsen, så den fylder mindre. Resultatet placeres med første adresse i <dest.adr.>. <dest.adr.> og <startadr.> kan evt. være den samme. Kompressionsgraden er som forklaret ved SA-kommandoen tidligere i manualen. Når kompressionen er færdig, vil længden af det komprimerede program blive vist på skærmen. Eksempel:

PACK 40000 4FFFF 50000 !200

Denne kommando komprimerer alle data mellem 40000 og 4FFFF (hexadecimalt angivet) og placerer de komprimerede data med første adresse i 50000 (hexadecimalt) med en kompressionsgrad på 200 (decimalt). Det er en god idé at skrive længden op, da du vil få brug for den ved gemning på diskette eller ved dekomprimering.

### *UNPACK <dest.adr.> <slutadr. for komprimerede data>*

Denne kommando fungerer omvendt af PACK. Det vil altså sige, at UNPACK-kommandoen dekomprimerer data, der er blevet komprimeret med PACK-kommandoen. <dest.adr.> angiver i dette tilfælde, hvor de dekomprimerede data skal placeres i hukommelsen. Her kan IKKE anvendes startadressen for de komprimerede data!

Du skal også angive slutadressen for de komprimerede data. Denne adresse kan du beregne ved at lægge startadressen for de komprimerede data og længden på dataene sammen. Hvis vi f.eks. antager, at længden af de komprimerede data i eksemplet ovenfor blev 1234, så vil det omvendte af eksemplet ved PACK blive:

UNPACK 40000 51234

### *COLOR <baggrund> <skrift>*

Sætter baggrunds- og skrift-farve på AR's monitorskærm ud fra AMIGA'ens palette på !4096 farver (nummereret fra 0-!4095). Farverne kan også ændres ved hjælp af preference-skærmen, som fås frem ved tryk på F3.

### *RCOLOR*

Denne kommando anvendes til at nulstille farverne på AR's monitorskærm i tilfælde af, at du har fået frembragt nogle upraktiske farver med COLOR-kommandoen.

### *TMS <adr.>*

Denne funktion er en slags notesblok, så du slipper for at skrive for meget op på papir, hvis der er et tal, f.eks. en adresse, du skal huske. Hvis du f.eks. skriver:

TMS A221

vil computeren huske den adresse, der giver flere liv i SuperOswald jvf. kapitlet om spilletræneren. Tallet vil også blive gemt med, hvis spillet gemmes på diskette med SA- eller SR-kommandoerne. Der er i AR plads til at huske 10 adresser på denne måde.

### *TMD <adr.>*

TMD sletter den angivne adresse fra notesblokken.

### *TM*

Viser de adresser/tal, der er skrevet op på notesblokken.

### *SPR <nr./adr.>*

Giver dig mulighed for at editere en angiven sprite (bevægelig figur fra spil e.lign.). Spriten kan enten angives med nummer eller adresse. Hvis du f.eks. trykker FREEZE i Workbench og skriver:

SPR 0

og trykker RETURN omkring tyve gange, vil du se pilens form komme frem på skærmen i form af en række tal - hvert tal svarer til en farve. Hvis du ønsker det, kan du med pile-tasterne (cursor-tasterne) ændre på spritens udformning. Husk at trykke RETURN i hver linie, hvor du har lavet en ændring. Læg mærke til, at det kun er tallene 0-3, der kan anvendes, idet der kun er fire farver til rådighed i samme sprite.

### *VIRUS*

Tjekker hukommelsen igennem for virus. Dette er normalt ikke nødvendigt, medmindre du har slået det automatiske virus-tjek fra i preference-skærmen (F3), da det automatiske virus-tjek vil advare dig, så snart en virus kommer i hukommelsen. Vær opmærksom på, at AR godt kan blive forvirret af andre virus-killere.

### *KILLVIRUS*

Hvis et virus blev fundet med VIRUS-kommandoen, kan det dræbes med KILLVIRUS-kommandoen.

# MASKINKODEMONITOREN

De kommandoer, der er knyttet til maskinkodemonitoren, er et enestående stykke værktøj til editering af den frosne hukommelse og dermed et frossent program. I dette tilfælde betragtes et program som frossent så snart, der er trykket på FREEZE-tasten. Mange af kommandoerne kan være meget nyttige for såvel den erfarne maskinkode-programmør som begynderen. Flere af kommandoerne (f.eks. D) er bidirektionale. Det vil sige, at man kan rulle skærmen både op og ned ved hjælp af piletasterne.

### *SETEXCEPT*

En let måde at komme af med de irriterende GURU beskeder. Samtidig får du en detaljeret beskrivelse af årsagen til fejlene, f.eks. ved SOFTWARE FAILURE.

### *COMP <startadr.> <slutadr.> <dest.>*

Sammenligner to hukommelsesområder. Det ene område går fra <startadr.> til <slutadr.>, det andet har startadresse i <dest.>. Alle forskelle ved sammenligningen vil blive rapporteret på skærmen.

### *SM <path> <navn>,<startadr.> <slutadr.>*

SM-kommandoen gemmer en angiven del af hukommelsen på diskette under <navn> i det directory, der evt. kan angives med <path>. Den del, der skal gemmes, angives med <startadr.> og <slutadr.>. Eksempel:

SM scroll,3021F 312EA

### *SMDC <path> <navn>,<startadr.> <slutadr.>*

Denne kommando gemmer en angiven del af hukommelsen i DC.B format. Idéelt for assembler-programmøren, idet et maskinkodeprogram herefter kan loades ind i mange assemblere, f.eks. DEVPAC, ARGASEM etc.

### *SMDATA <path> <navn>,<startadr.> <slutadr.>*

Svarer til SMDC, blot er det anvendte format DATA i stedet for DC.B.

### *A <adresse>*

Med A-kommandoen startes AR's indbyggede M68000 assembler. Herefter indtastes simpelt hen M68000 mnemonics. Alle standard 68000 instruktioner kan anvendes og så længe du skriver brugbare kommandoer kan du fortsætte indtastningen. For at afslutte trykkes ESC.

Denne manual er ingen vejledning i M68000 programmering, men en gennemgang af de særlige kommandoer, der er i AR. Derfor ligger det uden for manualen at forklare M68000-instruktionerne. Til gengæld findes der mange gode bøger om dette emne.

### *BS <adresse>*

Her er en helt unik kommando til fejlsøgning. BS indsætter et breakpoint på et angivent sted i hukommelsen. Når computeren når til denne adresse, vil AR's monitor-system blive startet op og AMIGA'en har stoppet programafviklingen på nøjagtig dét sted, du har angivet med <adresse>. Eksempel:

BS 49152
X

Når computeren når til adresse 49152 i programafviklingen, får du monitor-skærmen tilbage.

### *B*

Vis aktive breakpoints på skærmen.

### *BD <adresse>*

Slet breakpoint i <adresse>, der er sat der med BS-kommandoen.

### *BDA*

Sletter alle breakpoints, der er indsat med BS-kommandoen.

## X

X-kommandoen bruges til at genstarte et frossent program. Programmet vil gå videre som om intet var hændt. Enhver ændring, der måtte være lavet i registre eller hukommelse, vil blive bibeholdt. Store eller uheldige ændringer kan få computeren til at give uventede resultater eller i værste fald at crashe.

## C <1/2/adresse>

Med C-kommandoen startes Copper-assembler/-disassembleren. Efter C'et skal indtastes enten 1 eller 2 eller en adresse. AMIGA'en er udstyret med en Copper eller co-processor, der kan programmeres på samme måde som 68000-processoren, dog med et mere begrænset udbud af muligheder. Når du har trykket C med én af ovennævnte muligheder for efterfølgende parameter, vil Copper-instruktionerne for de pågældende adresser blive vist. Et tryk på RETURN vil acceptere instruktionen på den pågældende linie og flytte til den næste adresse i hukommelsen. For yderligere informationer om Copperen henvises til "Amiga Technical Reference Manual".

## D <adresse>

D-kommandoen starter AR's indbyggede disassembler. Disassembleringen foregår i M68000 m/c. Såfremt der ikke angives en adresse, vil disassembleringen starte i den adresse, hvor programmet blev fastfrosset. RETURN vil disassemblere den næste linie. Dette vil fortsætte, indtil funktionen stoppes med ESC.

## E <offset>

Viser chip-registrene binært. Disse værdier kan blive ændret ved hjælp af piletasterne. Tryk RETURN, når en linie er blevet ændret.

## F <tekststreng>,<startadr.> <slutadr.>

Søger hukommelsen igennem mellem <startadr.> og <slutadr.> for den angivne tekststreng. Teksten skal angives i anførselstegn. Eksempel:

F "COMMODORE",0 7FFFF

Denne kommando søger i hukommelsen mellem 0 og 7FFFF (hexadecimalt) efter teksten COMMODORE, men vil kun finde den, hvis den er skrevet med store bogstaver som angivet. Hvis du vil søge efter COMMODORE med både store og små bogstaver, skal du skrive FS i stedet for blot F.

## FA <adresse> <startadr.> <slutadr.>

Gennemsøger hukommelsen for MC68000-instruktioner, der adresserer <adresse>. FAQ i stedet for FA vil foretage søgningen hurtigere.

## FR <tekststreng>,<startadr.> <slutadr.>

Søg efter den angivne tekst, evt. med en offset-forskydning. Hvis et offset forekommer, kan "HALLO" se ud som "IBMMP", hvis offsettet er 1. I computerens hukommelse er hvert bogstav et tal. I visse tilfælde kan spil og andre programmer have "gemt" skærmtekster ved at lægge et tal til eller trække et tal fra på de tal, der hører til teksterne. Sådanne tilfælde vil blive afsløret med FR-kommandoen.

### *G <adresse>*

Hvis du blot trykker G uden adresse, vil det fungere på samme måde som X. G går nemlig ud af AR's monitor-system og starter derpå m/c-programafviklingen fra den adresse, der angives efter G. Hvis ingen adresse angives, anvendes den adresse, der blev frosset ved. Idéelt ved fejlsøgning og afprøvning af programmer, du netop har indtastet. Før programmet startes, sørger AR for, at alle registre sættes tilbage til den tilstand, de havde, før der blev trykket på FREEZE-knappen.

### *TRANS <startadr.> <slutadr.> <dest.>*

Kopierer en del af hukommelsen, angivet med <startadr.> og <slutadr.>. Første adresse i området, der kopieres til, er angivet med <dest.>. Denne kopi-funktion er intelligent. Det vil sige, at funktionen vil fungere selv om du angiver en <dest.> mellem <startadr.> og <slutadr.>.

### *WS <tekststreng>,<adresse>*

Skriver en tekststreng (en tekst i anførselstegn) i hukommelsen med det første tegn/bogstav i <adresse>. Eksempel:

WS "Testtekst",41259

### *M <adresse>*

Viser hukommelsen som hexadecimal værdier og giver mulighed for at ændre disse. Vil vise en linie ad gangen. De enkelte værdier kan ændres ved at flytte cursoren hen til den pågældende værdi og skrive oveni. Husk at trykke RETURN efter hver linie, der er blevet ændret i. Her kan kun angives hexadecimale værdier. Tryk ESC for at komme ud af editeringen.

### *N <adresse>*

Svarer til M-kommandoen med den forskel, at hukommelsen vises som ASCII-karakterer (tegn).

### *NO <offset>*

Med denne kommando fastlægges et tal (<offset>), der lægges til alle tal, før de vises på skærmen som tegn med N-kommandoen. Forklaringen på offset findes under FR-kommandoen. Hvis du f.eks. får "IBMMP" på skærmen med N-kommandoen, kan du indtaste kommandoen

NO 1

og derefter igen N-kommandoen. Denne gang skulle resultatet blot blive "HALLO". For at komme af med offsettet skrives

NO 0 (N + bogstavet O + nul)

Det nuværende offset kan ses ved at anvende NO uden offset.

### *NQ <adresse>*

Svarer til N-kommandoen, blot foregår det i et hurtigere format.

### *MEMCODE <startadr.> <slutadr.> <kode>*

Gå gennem en del af hukommelsen (<startadr.> til <slutadr.>) og koder det med koden <kode>. Kodningen foregår med 68000's EOR-instruktion. Dette betyder, at du enten selv

kan dekode det kodede område med EOR-instruktionen eller dekode med den samme MEMCODE-kommando igen. Eksempel:

MEMCODE 312FE 345A1 EA

Denne kommando koder området 312FE-345A1 med koden EA (alle angivelser er hexadecimale). For at få området tilbage til normale omstændigheder igen (dekodning) skrives atter:

MEMCODE 312FE 345A1 EA

### *O <tekststreng>,<startadr.> <slutadr.>*

Fylder det angivne hukommelsesområde op med <tekststreng>. Teksten skal være omgivet af anførselstegn ("").

### *R <register> <værdi>*

Viser registrene og giver mulighed for at ændre i dem. R-kommandoen alene uden parametre vil vise indholdet af alle 68000 registre fra det tidspunkt, hvor der blev trykket på FREEZE-knappen. Hvis du vil ændre i ét af registrene, skal du angive både <register> og <værdi>. Eksempel:

R D0 !1000

Vil placere 1000 (decimalt) i 32 bits data registret D0. ændringerne vil træde i kraft efter brug af X eller G.

### *W <register>*

Her er muligheden for at se indholdet af begge 8520 chips (even CIA / odd CIA). <register> skal angive hvor i området, regnet fra første adresse i CIA-området, at visningen skal starte. Indholdet i den pågældende adresse for begge CIA'er vises på samme linie. Eksempel:

W 4

vil vise indholdet af det fjerde register i både den første og den anden CIA.

### *Y <adresse>*

Svarer i høj grad til M-kommandoen, blot bliver hukommelsen med Y-kommandoen vist på binær form og altså ikke hexadecimalt. Med YS-kommandoen angives, hvor mange bits, du ønsker angivet pr. linie. Eksempel:

YS 8

vil "fortælle" Y-kommandoen, at den skal vise 64 bits på hver linie, idet 8*8=64. Adresserne vil naturligvis blive opdateret tilsvarende.

### *? <udtryk>*

En lille regnemaskine! Med ?-kommandoen kan du indtaste et aritmetisk udtryk eller blot et enkelt tal og få resultatet vist både binært, hexadecimalt og decimalt. Alle tre typer tal kan medtages med de respektive præfixer (intet for hexadecimalt, % for binært og ! for decimalt). Eksempel:

?8+7+!15*2

Dette vil give resultatet !60. Bemærk, at beregningen foregår fra venstre mod højre uden nogle prioriteter! De fire regningsarter kan bruges og angives med tegnene +, -, * og /.

# BOOT VIRUSKILLER

Du kan komme ud for, at AR's monitorsystem "pludselig" starter, når du sætter en diskette ind for at boote den. Dette skyldes, at AR har opdaget en virus på diskettens boot-block. For at undgå, at denne virus breder sig, bryder AR ind med det samme og skriver DANGER - DANGER o.s.v. foroven på skærmen.

For at dræbe den virus, der er på disketten, kan du i første omgang sørge for, at disketten ikke er skrivebeskyttet (man må ikke kunne se igennem hullet med plastictappen). Herefter trykker du F4, og når AR et kort øjeblik efter melder klar på skærmen, trykker du F5 for at fortsætte bootningen. Nu er disketten blevet kureret og har oven i købet fået installeret et anti-virus program, der tjekker hukommelsen igennem for virus, når disketten bootes.

Der er mange flere muligheder i BOOT VIRUSKILLEREN, men disse er for omstændige at komme ind på her; de er ikke nødvendige for den almindelige bruger. Imidlertid kan nærmere information om disse hentes på skærmen (på engelsk).

# SYSTEM INFORMATIONER

De fleste af kommandoerne i denne sektion giver komplekse system-informationer, som kun kan bruges af avancerede AMIGA programmører. Lad dig derfor ikke genere af det, hvis du ikke forstår hvad de er og betyder. Efterhånden som din forståelse af AMIGA'ens opbygning forøges, vil din forståelse af disse kommandoer ligeledes blive større.

### *AVAIL*

Viser på skærmen, hvor meget hukommelse, der er til rådighed i maskinen, såvel chipmem som fastmem. Idéelt til f.eks. tjek af, om en MAXIMEM eller POWERMEM 1,5 Mb er installeret ordentligt.

### *INFO*

Viser en oversigt over vigtige system informationer.

### *LIBRARIES*

Vil vise en liste over Execbase Libraries og de tilhørende adresser.

### *INTERRUPTS*

Giver en liste over de i øjeblikket aktive Execbase interrupts.

### *EXCEPTIONS*

Viser en liste over MC68000 exceptions samt hvor i hukommelsen de i øjeblikket peger hen.

### *EXECBASE*

Vis Exception og Interrupt vektorer.

### *RESOURCES*

Vis Execbase's Resource liste.

### *CHIPREGS*

Vis navn og offset på chipregistrene.

### *DEVICES*

Vis Execbase's device-liste.

### *TASKS*

Vis Execbase's task-liste.

### *PORTS*

Vis Execbase's port-liste.

# HURTIG GENNEMGANG

For dem, der gerne vil have en simpel forklaring på FREEZE-princippet, følger her et kort eksempel på en fremgangsmåde ved en FREEZEning. Her vises, hvilke kommandoer, der skal bruges. Nærmere forklaring må søges i selve manualen.

Først og fremmest skal AR sættes i computeren. Nu loades programmet, som du ønsker at gå ind i, på helt normal vis. Når det er i gang, og diskettedrevet er stoppet (den grønne lysdiode er gået ud), trykkes på FREEZE-knappen. Tryk nu evt. F3 og indstil, hvis der er noget, du vil ændre, f.eks. farver, autoboot, virus-detektor o.s.v. Gå tilbage til monitorskærmen med ESC.

Nu har du muligheden for at ændre i programmet med kommandoer som TRACKER, SCAN, P og spilletræner-kommandoerne.

Når du har ændret i programmet, som du selv vil det, vil du måske gemme det på diskette. Dette kan også gøres uden at have ændret noget i programmet. I begge tilfælde skal du bruge en formateret diskette med tilstrækkelig meget ledig plads på. Hvis du ikke har en sådan, kan du formatere en ny diskette med FORMAT-kommandoen (eller endnu bedre: FORMATV). Du kan gemme programmet på disketten med SA eller SR. Programmerne kan så en anden gang hentes ved hjælp af AR med LA eller LR kommandoen.

Hvis du gerne vil have, at det frossede program skal kunne startes uden AR i computeren, bruger du SLOADER og INSTALL kommandoerne. Når disketten skal bruges, sættes den simpelt hen i, når computeren lige er tændt. Efter et øjeblik kommer en blå skærm frem med en cursor. For at starte programmet skrives ALOAD efterfulgt af programmets navn (husk at trykke RETURN efter hver kommando).

God fornøjelse.

# TIPS & TRICKS

Før et program saves med SA-kommandoen (eller SR), er det en fordel at slå al hukommelse fra, der ikke anvendes, idet SA-kommandoen gemmer hele den til rådighed værende hukommelse på disketten. Dette vil altså spare både disketteplads og tid.

Det er ikke tilrådeligt at slå virus-tjekket fra i F3-menuen, også selv om du ikke regner med at computeren har virus. Virus er set selv på originale spildisketter!

Hvis spilletræneren ikke fungerer korrekt og umiddelbart opgiver at finde det søgte antal liv, kan det hjælpe at forsøge ved samtlige gennemsøgninger enten at lægge 1 til antallet af liv eller at trække 1 fra. Husk at angive et præfix foran tallet (som regel !). Dog lagrer nogle spil antallet af liv i BCD-formatet. I dette tilfælde skal der ikke angives noget præfix.

Sædvanligvis ligger spillerdata, som f.eks. liv, energi, score, brændstof o.s.v., ret tæt på hinanden i hukommelsen. Så når den ene værdi er fundet, er der store chancer for, at adresserne for de andre værdier er lige i nærheden.

Nogle adresser med spillerdata er inverterede, d.v.s. de tæller opad i stedet for nedad.

I monitorsystemet kan en anden skærm altid kaldes frem med F10. På denne skærm kan du efterlade noter eller andre ting, du skal huske. Den anden skærm bliver slettet, nær spilletræneren bruges. Den første skærm kan altid kaldes tilbage med SHIFT+F10.

Hvis du har et originalt program, der ikke ligger fuldstændigt i hukommelsen, det vil sige, at det hele tiden henter ekstra data fra disketten, egner det sig ikke rigtig til at blive kopieret med AR. Nogle gange kan en velfungerende backup imidlertid laves efter følgende opskrift:

Først kopieres hele disketten med Workbench eller med et godt diskettekopieringsprogram. Du kan også prøve med DCOPY-kommandoen. Læg denne kopi til en side. Start nu det originale program og FREEZE på det punkt, hvor du gerne vil have det til at starte op. Efter trykket på FREEZE-knappen, bruges SA-kommandoen til at save programmet på en formateret diskette med rigelig plads. Når backup-kopien nu loades, bruges LA-kommandoen i monitorsystemet til at loade programmet men ikke til at starte det. Byt nu disketten ud med den backup-kopi, som først blev lavet med det ordinære kopiprogram. Herpå bruges X eller G for at starte programmet, hvorefter det bør finde alle de oplysninger, det har brug for, på den først kopierede diskette. Af flere grunde er det ikke muligt at benytte denne metode med alle programmer, men det kan være meget nyttefuldt, hvor det er vigtigt at lave en sikkerhedskopi af et bestemt program. Samtidig vil det få programmet til at starte hurtigere op end originalen (i de fleste tilfælde).

***Vi håber, at denne manual har hjulpet dig godt i gang med brugen af dit nye AMIGA ACTION REPLAY MKII. Hvis du finder fejl i denne manual, er du mere end velkommen til at rapportere dette til din forhandler, så at det kan blive ændret i nye udgaver af manualen. Vi takker på forhånd og håber, at du vil få stor glæde og nytte af dit AMIGA ACTION REPLAY MKII i fremtiden.***